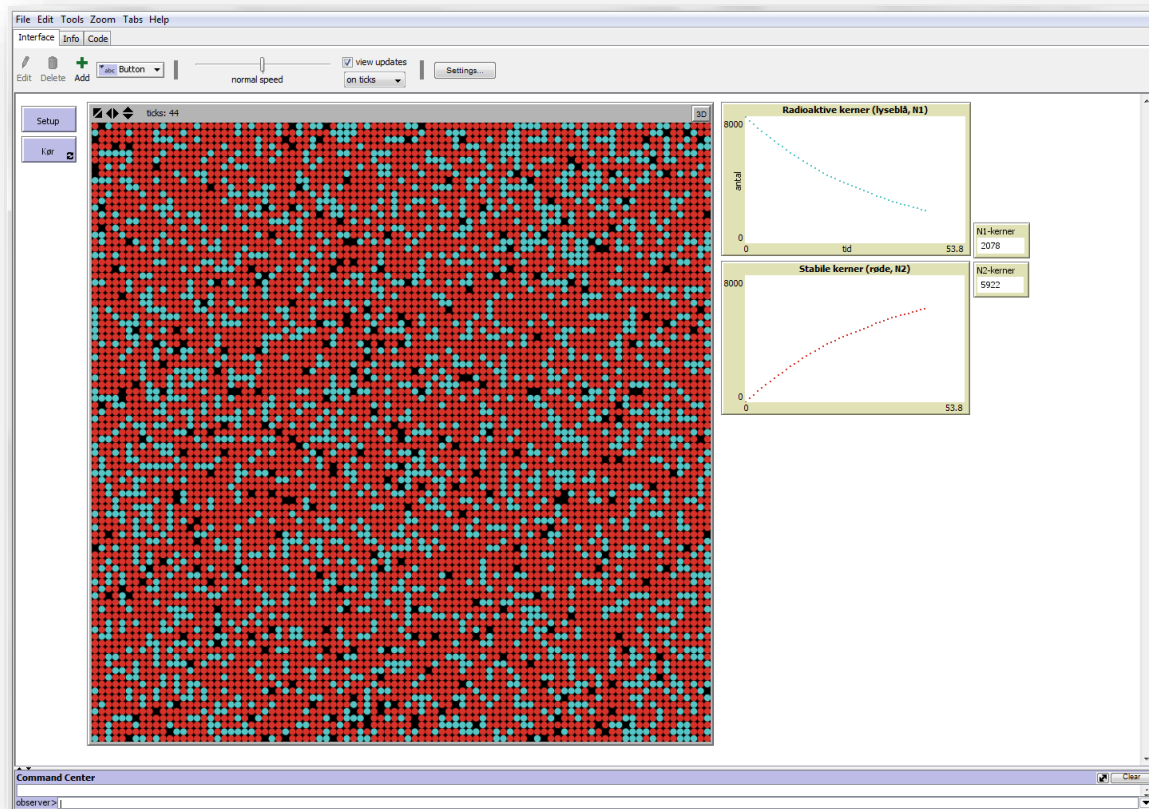


## VEJLEDNING TIL LÆREREN

I dette dokument er forløbet og elevernes arbejde beskrevet i lidt større detalje. Det vises hvordan de forskellige ”kodeopgaver” kan løses, og der gives et overblik over de mange filer på hjemmesiden.



### INDHOLD

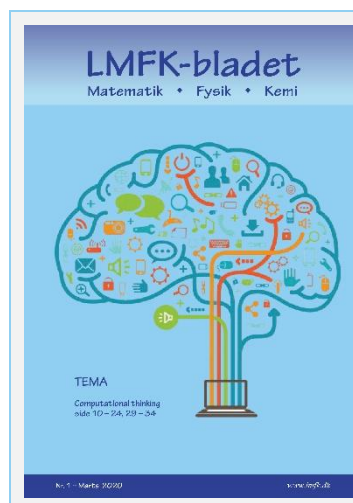
|                                                                             |    |
|-----------------------------------------------------------------------------|----|
| Kort om forløbet.....                                                       | 2  |
| Ændringerne i computermode llens kode (løsninger til ”kodeopgaverne”) ..... | 4  |
| Aktivitet 1c og 2b (Arbejdsark 1, opgave 3, 5 og 6).....                    | 4  |
| Aktivitet 3c (Arbejdsark 2, opgave 3b, 4d og 5h) .....                      | 5  |
| Aktivitet 4b (Arbejdsark 3 eller 4) .....                                   | 7  |
| Installation af NetLogo .....                                               | 9  |
| Overblik over dokumenter og filer .....                                     | 9  |
| Arbejdsark.....                                                             | 9  |
| NetLogo-filer.....                                                          | 10 |

## KORT OM FORLØBET

Forløbet er et induktivt forløb om henfaldsloven, hvor der arbejdes med computational thinking (CT) og med modelbegrebet i fysik. Eleverne præsenteres for den mikroskopiske model for henfald (at hver kerne har en given sandsynlighed for at henfalde hvert sekund) og ved hjælp af simuleringer med forskellige startværdier, finder de frem til henfaldsloven (den makroskopiske model). Herefter implementeres ændringer i koden af den udleverede computermodel, således at den kan bruges til at undersøge mere komplekse henfaldsprocesser. Der arbejdes med:

- Kulstof 14-metoden
- Henfaldskæder med udgangspunkt i Tjernobyl-ulykken eller en isotopgeneratorer

Undervejs trænes kompetencer indenfor *computational thinking* og færdigheder i *programmering* (at læse, forstå og skrive computerkode). Computermodellen er lavet i programmeringsmiljøet NetLogo og ser ud som vist på figur 1 på næste side.



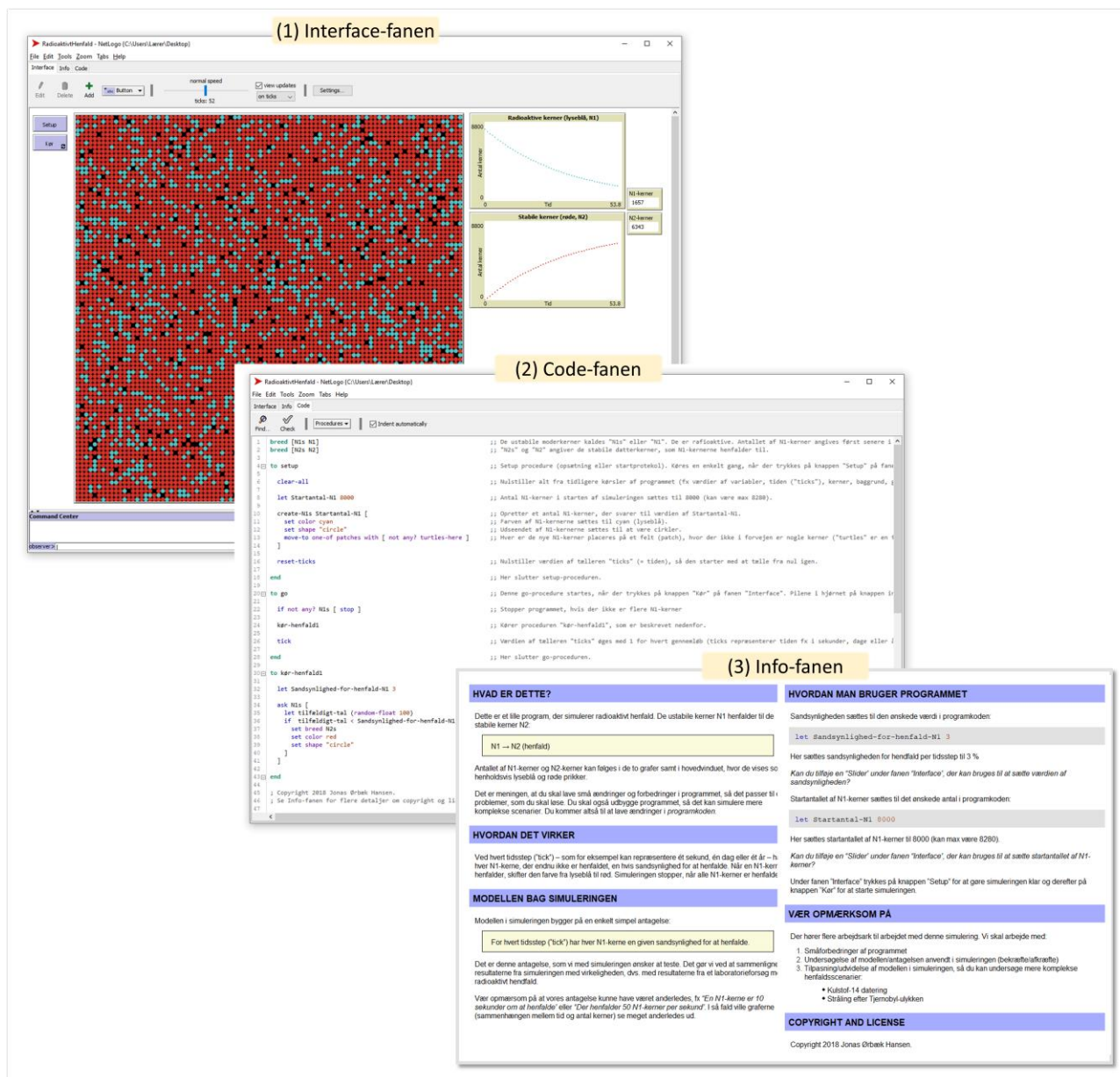
Forløbet er beskrevet i LMFK-bladet nr. 1 (februar/marts 2020) i artiklerne:

- Computational thinking i gymnasiefag (side 10-14)
- Computational thinking i fysikfaget – henfaldsloven (side 23-31)

LMFK-artiklerne kan findes her:

[http://www.lmfk.dk/index.phtml?sek\\_id=48](http://www.lmfk.dk/index.phtml?sek_id=48).

Computermodellen og de anvendte undervisningsmaterialer kan findes på <https://library.ct-denmark.org/lmfk/>. Her er også materialerne til en matematikforløb, der tager udgangspunkt i samme computermodel samt materialer til og beskrivelser af flere CT-forløb i fagene matematik, fysik, kemi, biologi, bioteknologi og NV. Desuden er der links til mere læsning, podcasts og videoer om *computational thinking i gymnasiefag*.



Figur 1: Computermodellen i NetLogo, som eleverne får udleveret (og efterfølgende skal lave ændringer i). Computermodellen består af (1) en brugerflade (Interface-fanen), (2) en kode, hvor de faglige modeller er skrevet ind (Code-fanen) og (3) en beskrivelse af simuleringen (Info-fanen). Eleverne skal under arbejdet implementere fagligt begrundede ændringer i koden og i brugerfladen (input/output).

## ÆNDRINGERNE I COMPUTERMODELLENS KODE (LØSNINGER TIL "KODEOPGAVERNE")

Nedenfor beskrives de ændringer i computermodellens kode, som eleverne skal igennem i de forskellige aktiviteter. Aktiviteterne henviser til Figur 1 på side 24 af LMFK-artiklen. Generelt gælder, at opgaverne kan løses på flere forskellige måder, så nedenstående skal blot ses som et eksempel, og kan eleverne finde på andre måder at løse opgaverne på, er det helt fint.

### Aktivitet 1c og 2b (Arbejdsark 1, opgave 3, 5 og 6)

I aktivitet 1c skal atomkernernes udseende (form og farve) ændres. I aktivitet 2b skal computermodellens startværdier ændres. For at løse disse opgaver, må eleverne analysere koden og finde de steder, hvor kernernes farve og form indstilles (markeret med grønt i figuren) og de steder, hvor startværdierne indstilles (markeret med blå).

```
breed [N1s N1]
breed [N2s N2]

to setup
  clear-all

  let Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    set shape "circle"
    move-to one-of patches with [ not any? turtles-here ]
  ]

  reset-ticks

end

to go
  if not any? N1s [ stop ]

  kør-henfald1

  tick

end

to kør-henfald1
  let Sandsynlighed-for-henfald-N1 3

  ask N1s [
    let tilfældigt-tal (random-float 100)
    if tilfældigt-tal < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
      set shape "circle"
    ]
  ]

end
```



Ændring af kernernes udseende



Ændring af simuleringens værdier

Et eksempel på løsning er givet i NetLogo-modellen [RadioaktivtHenfald\\_LÆRER\\_A1\\_Henfaldsloven\\_1.nlogo](#).

I ekstraopgave h skal den mikroskopiske model ændres i computermodellen for at se, hvorledes det påvirker den makroskopiske model. Den kan tages med, hvis man ønsker. Den mikroskopiske model kunne fx være, at en radioaktiv N1-kerne henfalder efter 50 sekunder, eller at sandsynligheden for at en radioaktiv N1-kerne henfalder afhænger af antallet af tilbageværende N1-kerner.

#### Aktivitet 3c (Arbejdsark 2, opgave 3b, 4d og 5h)

Opgave 3b i Arbejdsark 2 bør eleverne kunne løse relativt let på nuværende tidspunkt. Et eksempel på løsning er givet i NetLogo-modellen [RadioaktivtHenfald\\_LÆRER\\_A2\\_Kulstof14-datering\\_1.nlogo](#).

Det er ikke altid klart for mine elever, hvorfor indholdet af C-14 er konstant i levende organismer. Opgave 4 skal forsøge at hjælpe med forståelsen af dette. Eleverne skal ændre i koden af computermodellen, så den kan bruges til at undersøge, hvordan aktiviteten fra en radioaktiv prøve, der hele tiden får tilført nye radioaktive kerner med en konstant rate, udvikler sig som funktion af tiden. Det gør eleverne ved at studere kommentarerne i koden (under Code-fanen) og arbejde med rutediagrammerne over algoritmerne i computermodellen (se særskilt Power Point præsentation). Eleverne gjorde dette som lektie til lektionen, og vi gennemgik det i fællesskab i klassen, inden de blev kastet ud i opgave 4d. Derefter blev eleverne bedt om i grupper at diskutere strategier til at løse opgaven (hvordan problemet kan deles op i mindre dele). Nogle elever har brug for ekstra hjælp til at lave ændringerne. For disse elever er det en hjælp med en fælles løsningsstrategi, der kunne se ud som følger.

Eleverne skal:

1. Finde ud af hvordan man tilføjer radioaktive kerner (N1-kerner). Da der allerede er N1-kerner i simulationen til at begynde med, må der være et sted i koden, hvor de tilføjes. Dette sted skal eleverne finde og kodelumpen skal kopieres.
2. Finde ud af hvor i algoritmen denne kodelump skal tilføjes. Det kan gøres ved at studere koden, men det nemmeste er nok at bruge rutediagrammerne og markere dér, hvor der skal tilføjes nye N1-kerner. Dernæst findes dette sted i koden, og den kopierede kodelump sættes ind.
3. Rette det indsatte kodelykke til, så der tilføres netop 100 kerner per år.

Efter at have identificeret det stykke kode, der skal kopieres opstod der i nogle af grupperne uenighed om, hvor det er smartest at indsætte kodelumpen – før eller efter `kør-henfald1`, og før eller efter `tick`. Sunde diskussioner, der kun styrker deres forståelse af algoritmerne – og ultimativt af fysikken – i computermodellen.

At kunne dele et problem op i flere dele er en vigtig CT-kompetence. I figuren nedenfor er der med gult vist hvordan opgaven kan løses. Denne Løsning er også i NetLogo-filen [RadioaktivtHenfald\\_LÆRER\\_A2\\_Kulstof14-datering\\_2.nlogo](#).

**Oprindelig kode (udleveret):**

```

breed [N1s N1]
breed [N2s N2]

to setup
  clear-all

  let Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    set shape "circle"
    move-to one-of patches with [ not any? turtles-here ]
  ]

  reset-ticks

end

to go
  if not any? N1s [ stop ]

  kør-henfald1

  tick

end

to kør-henfald1
  let Sandsynlighed-for-henfald-N1 3

  ask N1s [
    let tilfældigt-tal (random-float 100)
    if tilfældigt-tal < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
      set shape "circle"
    ]
  ]
end
        
```

**Ny kode (mulig løsning):**

```

breed [N1s N1]
breed [N2s N2]

to setup
  clear-all

  let Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    set shape "circle"
    move-to one-of patches with [ not any? N1s-here ]
  ]

  reset-ticks

end

to go
  if not any? N1s [ stop ]

  kør-henfald1

  tick

end

to kør-henfald1
  let Sandsynlighed-for-henfald-N1 3

  ask N1s [
    let tilfældigt-tal (random-float 100)
    if tilfældigt-tal < Sandsynlighed-for-henfald-N1 [
      die
      ;; set breed N2s
      ;; set color red
      ;; set shape "circle"
    ]
  ]
end
        
```

Kode tilføjet (kopieret og rettet til fra andet sted)

Kode rettet til

Det røde i figuren ovenfor viser, hvordan koden kan ændres, så computermodellen ikke "fryser" eller kommer med en fejlmeddelelse, når den køres. Hvis eleverne blot laver de ændringer, som der er vist i NetLogo-filen [RadioaktivtHenfald\\_LÆRER\\_A2\\_Kulstof14-datering\\_2.nlogo](#), vil computermodellen komme med en fejlmeddelelse. Det skyldes, at der er flere kerner, end der er plads til i vinduet med røde og lyseblå kerner. Der tilføjes hele tiden nye N1-kerner, der langsomt henfalder til N2-kerner. Da vinduet viser både N1-kerner og N2-kerner, bliver der hurtigt for mange. Det skyldes, at der i programmet er skrevet ind, at kernerne ikke kan ligge oven i hinanden.

Eleverne må finde på noget for at løse dette problem. Efter at have diskuteret det i grupperne og studeret koden igen kom der forslag til forskellige løsninger, som de afprøvede:

- Første strategi var for mange (efter de havde fundet ud af hvad problemet var), at starte med færre N1-kerner eller lave et større område, som kernerne kan være på. Det er en god strategi - men den løser ikke det grundlæggende problem. Den forlænger bare processen, før fejlmeldingen kommer.
- Dernæst var der en gruppe der forsøgte at lave om i computermodellen, så kernerne kan ligge oven i hinanden. En fin løsning - der dog har den svaghed, at antallet af kerner i simuleringen vokser og



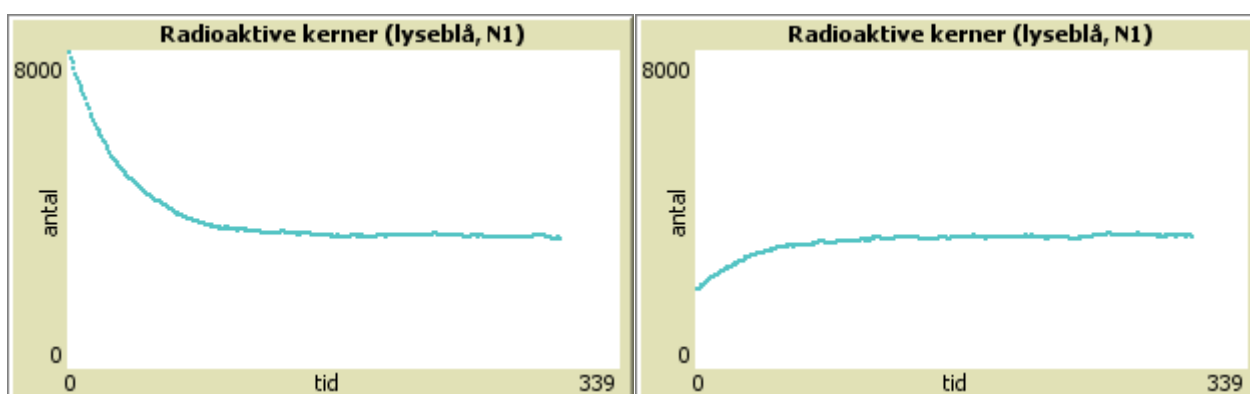
vokser, og at computeren før eller siden ikke kan håndtere de mange kerner (som gruppen fandt ud på den hårde måde med en "frossen" skærm).

- En anden gruppe undersøgte, om det ville hjælpe at indføre en tredje slags kerner N3. Det var dog et dødt spor.

En elegant løsning på problemet er kun at vise N1-kernerne i simuleringen og lade dem forsvinde, efterhånden som de henfalder til N2-kerner (i stedet for rent faktisk at lave dem om til N2-kerner). Alle grupperne havde brug for et hint og et puf i den rigtige retning, før de kom på denne løsning. For nogle var det nok at få spørgsmålet: "Er det vigtigt at vi kan følge antallet af N2-kerner i simuleringen?". Andre skulle hjælpes lidt mere på vej: "Hvad nu hvis vi kun følger N1-kernerne og lader dem forsvinde, efterhånden som de henfalder?".

En løsning på problemet – som blev udviklet lidt på tværs af grupperne – er vist med rødt i figuren ovenfor. NetLogo-modellen med løsningen er [RadioaktivtHenfald\\_LÆRER\\_A2\\_Kulstof14-datering\\_3.nlogo](#).

Ved at ændre på startværdierne i deres nye computermodel, kan eleverne se, at antallet af  $^{14}\text{C}$ -kerner i en organisme ender på et konstant niveau, der afhænger af halveringstiden for  $^{14}\text{C}$  og tilføjelsesraten, men ikke af startantallet af  $^{14}\text{C}$ -kerner:



Opgave 5h (ekstraopgave) er meget udfordrende, da koden skal laves grundlæggende om, og det er nok de færreste elever, der kan komme igennem denne. Det ekstra faglige udbytte ved opgaven er begrænset, men sidder der et par dygtige elever med mod på en kodeudfordring, så kan den eventuelt inddrages. Problemet, som eleverne render ind i, er, at der kommer alt for mange N1-kerner ( $\text{C-14}$ ) til at computermodellen kan køre. Løsningen er at lave en computermodel uden agenter (atomkerner). I stedet anvendes en global variabel, hvis værdi angiver antallet af N1-kerner. Et eksempel på løsning er givet i NetLogo-modellen [RadioaktivtHenfald\\_LÆRER\\_A2\\_Kulstof14-datering\\_4.nlogo](#).

#### Aktivitet 4b (Arbejdsark 3 eller 4)

I aktivitet 4b skal eleverne ændre i koden og interfacet af computermodellen, så den kan anvendes til at simulere end henfaldsproces, hvor datterkernerne fra det første henfald er radioaktive og henfalder igen. Eleverne skal bruge den nye computermodel til at undersøge, hvordan antallet af N2-kerner (det midterste stof i henfaldskæden) ændrer sig som funktion af tiden. Det gøres med udgangspunkt i Tjernobyl-ulykken (arbejdsark 3) eller nuklearmedicin (Arbejdsark 4). Eleverne skal igennem mange af de samme analyser og

overvejelser som i aktivitet 3c. Nedenfor er vist en mulig løsning (kode + interface) for eksemplet med Tjernobyl-ulykken. Løsningen for henfaldskæden i nuklearmedicin er identisk – dog med andre værdier og enheder for henfaldskonstanterne.

Eksempler på løsninger er givet i NetLogo-modellerne *RadioaktivtHenfald\_LÆRER\_A3\_Tjernobyl\_1.nlogo*, *RadioaktivtHenfald\_LÆRER\_A4\_Nuklearmedicin\_1.nlogo* og *RadioaktivtHenfald\_LÆRER\_A4\_Nuklearmedicin\_2.nlogo*. I sidstnævnte er der tilføjet to knapper på Interface til eluering af isotopgeneratoren. Den ene knap eluerer en enkelt gang, idet den trykkes. Den anden knap gør, at der elueres hver 24. time.

### Oprindelig kode (udleveret):

```
breed [N1s N1]
breed [N2s N2]

to setup

  clear-all

  let Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    set shape "circle"
    move-to one-of patches with [ not any? turtles-here ]
  ]

  reset-ticks

end

to go

  if not any? N1s [ stop ]

  kør-henfald1

  tick

end

to kør-henfald1

  let Sandsynlighed-for-henfald-N1 3

  ask N1s [
    let tilfældigt-tal (random-float 100)
    if tilfældigt-tal < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
      set shape "circle"
    ]
  ]

end
```

 Kode tilføjet (kopieret og rettet til fra andet sted i koden)

 Kode rettet til

### Ny kode (lavet af elev):

```
breed [N1s N1]
breed [N2s N2]
breed [N3s N3]

to setup

  clear-all

  let Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    set shape "circle"
    move-to one-of patches with [ not any? turtles-here ]
  ]

  reset-ticks

end

to go

  ;; if not any? N1s [ stop ]

  kør-henfald1
  kør-henfald2

  tick

end

to kør-henfald1

  let Sandsynlighed-for-henfald-N1 55.45

  ask N1s [
    let tilfældigt-tal (random-float 100)
    if tilfældigt-tal < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
      set shape "circle"
    ]
  ]

end

to kør-henfald2

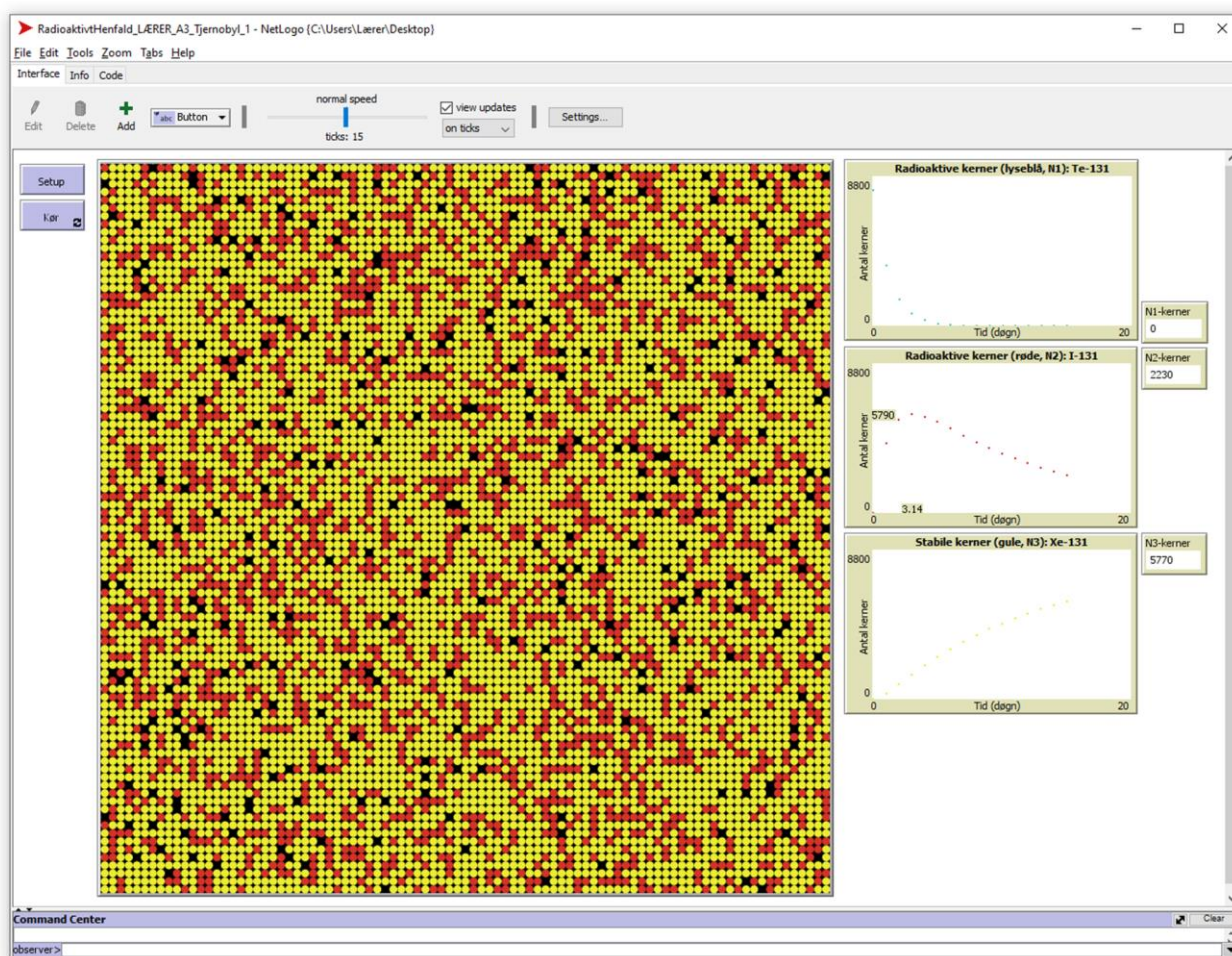
  let Sandsynlighed-for-henfald-N2 8.664

  ask N2s [
    let tilfældigt-tal (random-float 100)
    if tilfældigt-tal < Sandsynlighed-for-henfald-N2 [
      set breed N3s
      set color yellow
      set shape "circle"
    ]
  ]

end
```

Udkommenteret, så simuleringen bliver ved at køre, selvom der ikke er flere N1-kerner tilbage





## INSTALLATION AF NETLOGO

Du kan installere NetLogo fra <https://ccl.northwestern.edu/netlogo>. De NetLogo-modeller, der omtales i LMFK-artiklerne, kan findes på [library.ct-denmark.org/lmfk](http://library.ct-denmark.org/lmfk).

## OVERBLIK OVER DOKUMENTER OG FILER

### Arbejdsark

Der blev anvendt nedenstående arbejdsark. De nævnte aktiviteter henviser til Figur 1 på side 24 af LMFK-artiklen.

- *Arbejdsark 1 - Henfaldsloven.docx*: Aktivitet 1b - 2b
- *Arbejdsark 2 - Kulstof 14-datering.docx*: Aktivitet 3a - 3c
- *Arbejdsark 3 - Tjernobyl-ulykken.docx*: Aktivitet 4a - 4b
- *Arbejdsark 4 - Nuklearmedicin.docx*: Dette arbejdsark kan anvendes i stedet for Arbejdsark 3 i aktivitet 4a - 4b

## NetLogo-filer

Den NetLogo-fil som blev udleveret til eleverne:

- *RadioaktivtHenfald.nlogo*

Løsninger til opgaverne som de kunne se ud, efter eleverne har arbejdet med dem:

- *RadioaktivtHenfald\_LÆRER\_A1\_Henfaldsloven\_1.nlogo*: Mulig løsning til Arbejdsark 1, opg. 3 og 5.
- *RadioaktivtHenfald\_LÆRER\_A2\_Kulstof14-datering\_1.nlogo*: Løsning til Arbejdsark 2, opg. 3b.
- *RadioaktivtHenfald\_LÆRER\_A2\_Kulstof14-datering\_2.nlogo*: Første trin i løsningen af Arbejdsark 2, opg. 4d (OBS: computermodellen virker endnu ikke).
- *RadioaktivtHenfald\_LÆRER\_A2\_Kulstof14-datering\_3.nlogo*: Mulig færdig løsning til Arbejdsark 2, opg. 4d (nu virker computermodellen).
- *RadioaktivtHenfald\_LÆRER\_A2\_Kulstof14-datering\_4.nlogo*: Mulig løsning til Arbejdsark 2, opg. 5h.
- *RadioaktivtHenfald\_LÆRER\_A3\_Tjernobyl\_1.nlogo*: Mulig løsning til Arbejdsark 3, opg. 3a og 3b.
- *RadioaktivtHenfald\_LÆRER\_A4\_Nuklearmedicin\_1.nlogo*: Mulig løsning til Arbejdsark 4, opg. 3a og 3b.
- *RadioaktivtHenfald\_LÆRER\_A4\_Nuklearmedicin\_2.nlogo*: Her er tilføjet to knapper på Interface til eluering af isotopgeneratoren.

NetLogo-fil med aktivitet (ikke anvendt i min afprøvning – kan nemmere sammenlignes med eksperimentet (men koden er længere):

- *RadioaktivtHenfald\_Aktivitet.nlogo*